New dynamic events-based public transport router for agent-based simulations

Date of submission: 2013-08-1

Sergio Arturo Ordóñez Medina
Future Cities Laboratory (FCL), Singapore ETH Centre (SEC), 138602 Singapore
phone: +65-9298 7627
fax: +41-44-633 10 57
ordonez@ivt.baug.ethz.ch

Alex Erath
Future Cities Laboratory (FCL), Singapore ETH Centre (SEC), 138602 Singapore
phone: +65-8409 8023
fax: +41-44-633 10 57
alexander.erath@ivt.baug.ethz.ch

Words: 5984 words + 6 figures = 7484 word equivalents
ABSTRACT

Public transport route choice is an open problem studied for many years and it is well regarded that many factors influence the choice of stages making up a public transport trip. In addition to factors such as travel time, fare, service frequency, number of transfers and transfer waiting time, the focus of research shifted more and more to dynamic factors such as crowdedness and travel time variability. This paper describes a method to simulate public transport route choice for a whole population in a city or even a country taking into account all factors mentioned above using the MATSim framework, a multi-agent and activity-based transport simulator. A new public transport router is proposed using a time dependent shortest path algorithm in a graph with links representing in-vehicle travel, waiting and walking (from activity locations to public transport facilities and between public transport facilities). The cost of the path can be specified to depend on the fares, travel times, occupancy levels, waiting times and number of transfers. In addition, the agent-based nature of the simulation model allows to define agent-specific parameters for each attribute to account for preference heterogeneity within the agent population. Simulated dynamic effects such as bus bunching and overcrowded services lead to additional waiting and travel times, compared with the public transport schedule. Within the MATSim evolutionary algorithm, this information is saved after each simulation of the same day, and used to re-route agents. The router is implemented and tested for Singapore, whose modeled public transport system features bus and subway. As in most major cities, the model shows that the stability and schedule reliability of bus services can at times only hardly be maintained, while crowdedness at the peak hours is an issue for both bus and mass rapid transit and light rapid transit. We illustrate how the new approach allows agents to adapt to these dynamic complexities. The new approach, along with the original scheduled-based router of MATSim are compared for computation time, as well as for agent learning rate, i.e. the time needed to reach a particular solution state. With a better capacity to model observed public transport route choice, more than 10% of improvement in computation time per simulation iteration, and needing much smaller number of iterations to reach user equilibrium, the proposed alternative results a Pareto-dominant improvement.
INTRODUCTION

Urban mobility is a complex system where the decisions and actions of millions of actors are related in time and space. More specifically in public transport route choice, decisions and actions of a particular user depend not only on their own preferences (value of time, crowd avoidance, willingness to pay) but also on the decisions and actions of many other public transport users, operators, authorities and even private transport users, as they can share the same infrastructure.

MATSim is a software framework to simulate transport demand and supply interactions for millions of agents, each representing an individual person. MATSim includes a full simulation of public transport, where thousands of public transport vehicles can share roads with other vehicles, i.e. buses interact with cars and vice versa. Each agent has a transport demand represented by a chain of activities it has to perform in one day at different times in different places. The decisions on how to travel between places to perform these activities are made before the mobility simulation as a plan. Models to decide the route, start time, mode and/or destination of the journeys per person are included in MATSim and they can depend on their socio-demographic characteristics. However, as in reality decisions can change according to the effects others produce, MATSim executes an evolutionary algorithm to optimize utility of agents. The same day is executed hundreds of times, making changes in some agent planned journeys each time, remembering good decisions or forgetting bad decisions. More specifically, the experience of each agent in relation with the others is scored according to utility functions and only good plans are remembered. The process reaches a user equilibrium where no agent can improve their score any further by making any of the allowed changes to their plans.

This work is focused on public transport route choice. The current implementation of MATSim implements a schedule-based public transport router (SBR). It means, when an agent needs a route for a given start time, origin and destination, the SBR finds the shortest path in a schedule-based network assuming public transport vehicles are always on time and always have space. Within the mobility simulation a given vehicle can arrive early or late and/or it can be full, thus not allowing additional passengers to board. Hence, when a bad experience happens, the agent obtains a bad score and it is possible this plan will get replace with a more favorable one during the iterative learning process. The problem is that if the agent tries to find a new route for the same start time, origin and destination, the shortest path in the public transport scheduled network is going to be the same, and agents can not improve their experiences by changing route.

To address this shortcoming, a new events-based public transport router (EBR) is proposed, modeled, implemented and tested. It takes the given schedule as a base for the first iteration, but updated information on travel times, occupancy of the public transport vehicles, and waiting times is propagated between subsequent iterations. Thus, when executions of the same day are performed, new routes can be generated for the same start time, origin and destination, because the system is remembering delayed bus services (longer travel times), or train services where the vehicle arrives full (longer waiting times). However, the network within agents are routed needs a new topology to account for such variables. This approach allows then to account for emergent phenomena: in situations where overcrowded vehicles prohibit boarding, it makes sense for some agents to travel for a few stops in the outbound direction to transfer to a vehicle with sufficient capacity heading inbound to be able to board. Although more memory is needed, similar or even better computation times are achieved when shortest path calculations are performed due to the simpler network topology. Furthermore, to achieve user equilibrium...
requires a significantly fewer number of iterations.

The paper is organized as follows: first previous studies are reviewed, followed by a more
detailed description of the public transport simulation in MATSim and its evolutionary algorithm
for optimization. The proposed public transport network is explained in detail highlighting
the structures designed to save public transport vehicle travel times and public transport users
waiting times. The Singaporean scenario is described with which the EBR is tested, measured
and compared with the SBR. Finally conclusions are presented and future work is proposed.

LITERATURE REVIEW

Transit assignment is a field widely studied for more than 50 years. The importance of taking
into account waiting times in public transport networks was published by Dial (3). From
there frequency-based models like (4) and (5) where the assignment is not fixed to a run of a
fixed service and schedule-based models like (6) and (7) have been advance in the direction of
giving alternatives for the route choice dependent on congestion and low-reliable services and
crowdedness. Another interest have emerged since the late 1980s (8, 9) where hyperpaths
are assigned to users instead of single paths. Recent works like the one presented by Trozzi et al.
(10) where a dynamic frequency-based assignment with hyper-paths is modeled try to include
effects of queues, congestion and capacity constraints in optimization models. They highlight
the importance of running these models for large scale scenarios. In general the objective of such
work is to improve the classic traffic assignment strategy. These methods minimize aggregated
objective functions in aggregated representation of the space with aggregated representations
of demands and supply. It results very complex to run dynamic large-scale scenarios including
preference heterogeneity among travelers, interactions between them, fine location analysis, or
interaction effects when public and private transport users share the infrastructure.

On the other hand, simulation tools have been designed to model the complex relation be-
tween a full detailed public transport system (the supply) and time-dependent demands. In recent
works Chen and Chen (11) use stochastic simulations to assess service reliability with similar
conditions of the ones described before (high frequency in bus route services, dynamic demand,
capacity constraints, etc). When agents are used for the simulation personalized conditions can
be given for operators and public transport users. Oded (12) shows how many complex systems
like traveller perception and decisions, centralized controls, transit operations, among others,
can be connected in a multi-agent simulation to obtain traffic assignment. Neumann and Nagel
(13) present how a large-scale multi-agent public transport simulation can be used to easily test
scenarios to design paratransit services or identify areas with insufficient supply.

USER EQUILIBRIUM AND PUBLIC TRANSPORT IN MATSIM

MATsim is a platform to simulate transport demand and supply interactions allowing for large-
scale scenarios where millions of agents representing people interact. For each agent, a daily
activity plan is assigned representing the sequence of activities it has to perform at different
times and at different locations within a specific period of time (in general one day). MATSim
utilises an evolutionary algorithm to reach a steady state. That is, the same day is simulated
many times, where a fraction of the agents mutate their plans after each iteration. There are
many ways to modify the plans. They can change the departure time, the travel mode of a
sub-tour, or the location of a given type of activity, among others. This work is focused on the
modification of the route, more specifically for public transport users. The utility of the day is
measured for each agent in each iteration using the function described in Charypar and Nagel
MATSim includes a full implementation of public transport. On the transportation supply side the system is represented by stop facilities and transit lines. Several routes belong to each line. Each of these transit routes holds the information of the sequence of stop facilities with the expected arrival and departure offsets, the sequence of links in the road network a vehicle of this route has to follow, and the departure times of all the services of the route. As the links that public transport vehicles have to follow belong to the road network that private vehicles use in the simulation, public transport vehicles are affected by congestion. The dwelling process can be modeled in two ways: the simple approach just calculates the time a vehicle has to be stopped according to the number of passengers, type of vehicles, number and configuration of vehicle entrances and exits, and vehicle occupancy; a more fine-grained approach simulates a queue agents use to enter the vehicle. For bus stop facilities, the availability of bus bay can be specified to account whether a bus is obstructing a link for bypassing cars during the dwelling process. A same vehicle can be scheduled to perform several services; if it is late, the next service won’t be able to start. Thus, the level of detail of the public transport module allows to simulate phenomena such as early or late services, crowded vehicles, bus or train bunching or even long waiting times due to fully loaded vehicles. Figure 1 shows how MATSim can reproduce bus trajectories for a specific line in Singapore compared with the extracted data from the public transport smart card. The figure shows as well bus-bunching cases due to the size of the line, congestion and long dwelling times.

To route public transport trips, MATSim implements a scheduled-based public transport router (SBR). Given a public transport schedule it generates a spatio-temporal network with the topology shown in the Figure 2(a). Each node of the network represents a stop-route relation. Nodes from the same stop facility are located at the same point. There are two types of links: in-vehicle links generated for each route between two consecutive stop facilities, and transfer links generated between nodes denoting walking distance/time (in particular transfer links between nodes of the same stop facility are generated). In-vehicle links have a time-dependent disutility which is composed of the scheduled waiting time to take the next service, and the scheduled in-vehicle time, derived from consecutive stop offsets. Transfer link disutilities depend on a
arbitrary transfer penalty and the walking distance and/or time between corresponding nodes. For a given start time, origin and destination, the SBR selects a set of nodes within a minimum walking time from the origin and a set of nodes near the destination. Fares can be added as costs to the links according to the particular fare system. A multi-node Dijkstra algorithm is applied to find the shortest path between the set of origin nodes and the set of destination nodes with initial walking disutilities. Time dependent link costs are consulted while the algorithm advances in time and space. The resulting path with the lowest disutility is then converted to a sequence of trips for the agent.

Thereby public transport agents receive a planned route that may or may not be successful in the simulation. MATSim measures the performance of the plan with the mentioned scoring function which takes into account the same types of disutility used to create the route, but now using the actual in-vehicle time, waiting time, transfer cost, fares (that might deviate from a purely link based implementation), and walking time experienced in the simulation. Although the agent can ‘realize’ from the plan score that the plan was not successful or inefficient (i.e. because the bus came early or the train was full); if the start time, origin and destination don’t change for the next iteration the best path returned by the SBR won’t change either. As the structure and the disutilities of the network are generated purely based on the public transport schedule, which never changes, the agent won’t be able to find a better route.

EVENTS-BASED PUBLIC TRANSPORT ROUTER

A new public transport router (EBR) was developed for MATSim with the objective of modeling more realistic public transport route choice, where people learn over time that the transit vehicles are not always on time, do not always have sufficient space to allow for boarding a vehicle and trips with more comfort can be preferred.

Network topology

Figure 2(b) shows the structure of the proposed public transport network. Inspired by the network designed by Spiess and Florian (9) this implementation has two types of nodes. The first type represents a stop facility (green-black squares) as point in space while the second type (yellow-red dots) represents stop-route relation which can be seen as a physical or virtual platform for each line which passes a particular stop facility. For example different platforms in a metro system need to be modeled as different stop facilities because different services arrive at each platform and walking paths are needed to change from one platform to another. For bus stop facilities, they represent virtual platforms as in reality buses of different lines serving the same bus stop will normally use the same physical infrastructure e.g. a bus bay. To connect those nodes, there are four types of links. The in-vehicle links join two consecutive stop-route nodes in the direction of the correspondent route. The boarding links connect a stop node with each corresponding stop-route node. The alighting links are the opposite, so they connect the stop-route nodes with their corresponding stop node. Finally walking links connect a stop node with all the other stop nodes located within a walkable distance.

Link costs

Each link in this network has a related time-dependent disutility function. Different costs are saved for different times in the day for a given time bin (currently 15 minutes). In-vehicle link disutilities depend on the vehicle travel time, the travel distance, the level of occupancy and a fare rate if this system is distance-based. Boarding links disutilities depend on waiting
FIGURE 2  Comparison of the network topologies of the schedule-based transit router (a) and the new events-based transit router (b).
times, a transfer cost, and a fixed fare if this system is entry-based; thus it is possible to relate specific stop-route waiting times to these links. As the first waiting link is not a transfer, this cost has to be subtracted from the whole path cost, but this detail does not affect the shortest path calculation. Alighting links don’t have any cost associated, but it is possible to relate a fare to them. Finally, walking links depend on the walking travel time and distance. Equation (1) shows linear versions of these functions used in this model assuming a distance-based fare system.

\[
C_{iv}(t) = (\beta_{iv} \times t_{iv}(t))(1 + g(p_{oc}(t)) \times \beta_{vd} \times l_{iv}) + f_{iv} \times l_{iv} \\
C_{bo}(t) = \beta_{wt} \times t_{wt}(t) + c_{tr} \\
C_{al}(t) = 0 \\
C_{wk}(t) = \beta_{wk} \times t_{wk} + \beta_{wd} \times l_{wk} \\
C_{path}(t) = \sum C_{iv}(t') + \sum C_{bo}(t') + \sum C_{al}(t') + \sum C_{tr}(t') - c_{tr}
\]

\[C_{path}: \text{Total cost of the path.}\]
\[C_{iv}: \text{Cost of one in-vehicle link.}\]
\[C_{bo}: \text{Cost of one boarding link.}\]
\[C_{al}: \text{Cost of one alighting link.}\]
\[C_{wk}: \text{Cost of one walking link.}\]
\[\beta_{iv}: \text{Personalized cost per unit of time travelling in a vehicle.}\]
\[\beta_{vd}: \text{Personalized cost per unit of distance travelling in a vehicle.}\]
\[\beta_{wt}: \text{Personalized cost per unit of time waiting in a stop.}\]
\[\beta_{wk}: \text{Personalized cost per unit of time walking.}\]
\[\beta_{wd}: \text{Personalized cost per unit of distance walking.}\]
\[c_{tr}: \text{Personalized cost for make a transfer}\]
\[f_{iv}: \text{Vehicle dependent fare rate by distance travelled}\]
\[t_{iv}(t): \text{In-vehicle travel time (from Stop-stop travel times structure).}\]
\[t_{wt}(t): \text{Waiting time (from Stop-route waiting times structure).}\]
\[t_{wk}: \text{Walking time.}\]
\[l_{iv}: \text{In-vehicle distance.}\]
\[l_{wk}: \text{Walking distance.}\]
\[p_{oc}(t): \text{Occupancy level in the in-vehicle link (from Route-stop occupancy structure).}\]
\[g(p): \text{Simplified function of how occupancy level increases the cost (Equation 2)}\]

\[
g(p) = \begin{cases} 
0 & \text{if } p \leq p_{sit} \\
r_{sta} \times p + b_{sta} & \text{if } p_{sit} < p < 1 \\
b_{full} & \text{if } p = 1
\end{cases}
\]

\[p_{sit}: \text{Occupancy level when no more seats are available.}\]
$r_{sta}, b_{sta}$: Parameters of percentage increase in discomfort from standing in the vehicle.

$b_{full}$: Maximum percentage increase when the vehicle is full.

### Shortest path algorithm

To find a public transport route between an origin and a destination for a given time in the day the applied method is the same as currently implemented in MATSim: first, the algorithm looks for the stop-nodes within a walkable distance from the origin and from the destination. An initial cost is associated with each of these stop-nodes according to access and egress walking times. Then, starting from all the origin-stop-nodes with a given access cost, a multi-node time dependent Dijkstra algorithm finds the shortest path, to the destination-stop-nodes with related egress costs. Thus, the path determines the best origin-destination combination as well. The algorithm is time-dependent because it takes into account that time is advancing while it proceeds through the path; thus, different costs are obtained from the links while time is advancing. The total disutility of this path is compared with the cost of a full walking trip. If the cost is less, the path is converted to a sequence of stages: in-vehicle stages for each in-vehicle link in the path and walking stages for each walking link. Boarding and alighting links are ignored for this conversion.

### Structures to save travel times, waiting times and vehicle occupancy

The mobility simulator of MATSim generates atomic units of information called ‘events’, which describe state changes for each person e.g. boarding and each vehicle e.g. entering and leaving a link during the course of the simulation. Event handlers can be added to the simulation engine to process events and obtain statistics of the simulation. The objective of this work is to save information of the public transport experience in one simulation to find better public transport routes for the agents in the next iteration. This feedback mechanism is already implemented in MATSim for private transport. The car router uses time-dependent travel times of each link saved in a previous iteration to calculate better routes in the road network, by changing the costs of the links. To allow the EBR to learn from the previous iteration, information about a) stop-stop travel times, b) stop-route waiting times and c) route-stop-stop vehicle occupancy, is required.

### Stop-stop travel times

In order to account for delays of public transport vehicles, the travel time between consecutive stops must be saved. Two stops are consecutive if they are consecutive at least for one public transport route. A first option is to use the mentioned travel times structure that saves time-dependent travel times for each link in the road network. As the road links a vehicle has to follow between two consecutive stops are known these travel times can be summed. The problem is that this structure takes into account all the vehicles in the network, and particularly in the links where public transport stops are located, travel times of cars and buses are very different. Consequently a special structure was implemented to save these stop-stop travel times. The structure averages all the public transport vehicle times from one stop to a consecutive one during a certain time bin in the day. More specifically, each value comprises the time since the vehicle arrived at a certain stop until the vehicle arrives at the consecutive stop, denoted in the simulation by consecutive ‘VehicleArrivesAtFacility’ events. This means that the first stop dwelling time and the second stop queue time (if the vehicle has to queue before the bay or platform is available) are included. These stop-stop times are the principal component of the in-vehicle link disutilities. When an agent is routing the first in-vehicle link of each trip (as all
the other in-vehicle links) it will count the full dwell time, hence it assumes the case in which it enters the vehicle first.

Stop-route waiting times

Waiting times are a fundamental aspect in public transport route choice. Waiting times can be long due to vehicle delays (i.e. due to the stop location), or when public transport vehicles of one or several consecutive services are full (i.e. due to the route demand and stop position within the route). For that reason waiting times are saved for each stop-route relation. Similarly the structure averages all the agent waiting times in a certain stop for a certain route during a certain time bin in the day.

More specifically each value comprises the time since the agent arrives to the public transport stop until it enters the vehicle, denoted in the simulation by consecutive ‘AgentArrivesToFacility’ and ‘PersonEnterVehicle’ events. These waiting times are the principal component of the boarding link disutilities.

Route-stop occupancy

To account for the occupancy level allows to model routing decisions where people take in terms of travel times longer routes in order to feel more comfortable in emptier vehicles e.g. valuing a higher chance to travel seating. Occupancy depends on the demand of a certain route and the position of the stop within the route. The occupancy is assumed constant between two consecutive stops. When a vehicle departs from a certain stop (denoted in the simulation as ‘VehicleDepartsFromFacility’ event) this structure averages the occupancy level with the other vehicles of the same route that departed from the same stop during the same time bin. As there are few vehicles for each time bin it is very likely not to find observations for a certain time bin. In this case the structure returns the value of the next time bin where at least one observation is found of the corresponding stop and route.

SINGAPORE SCENARIO

The MATSim Singapore model has been developed at the Future Cities Laboratory since 2011. The model includes a synthetic population created with the Household Interview Travel Survey of 2008 and aggregated information from the 2010 National Census. More than 100,000 facilities are included covering residential, business, industrial, shopping to recreational locations on the spatial resolution of individual buildings. More than 2 million motorized travellers are assigned to residential locations, working or studying places and other types of activity facilities. The road infrastructure was modeled with a high resolution network developed by NAVTeq, version Q1 2011, with more than 60,000 links and 40,000 nodes. The model includes Singapore’s dynamic road pricing scheme, special trips (freight and tourism) which are only subject to rerouting and the public transport system explained as follows.

Public transport system

The Singaporean public transport system in 2011 included 4,783 stop facilities, composed of 4,547 bus stops and 236 rail platforms. The rail platforms are distributed among 112 stations. The system comprises 331 bus lines, each line is composed of several routes according to the direction, start stop, end stop and schedule. In total there are 770 bus routes. There are 10 rail lines with 33 rail routes, and 95,532 buses of 20 different types and 2943 trains depart during a
common weekday. With the Google Transit Feed Specification of the whole system provided by
the Land Transport Authority of Singapore (LTA) and semi-automatic processes to map-match
the system with the road network, the MATSim Singapore model includes all the quantities
mentioned. The model includes the specification of roads where buses have an exclusive lane
and stops without bays where the traffic is blocked during dwelling processes. Following the
study developed by, dwelling process times are stochastic in this model, with higher standard
deviations when more people are involved. As the working scenario is prepared to simulate
25% of the people, every vehicle will be simulated, but each vehicle’s size and capacity will be
reduced to a fourth.

**Simplified public transport simulation**

Since in this work is only concerned about public transport routing and simulation and in order
to perform comparisons between the SBR and the EBR with less computation requirements, an
innovative strategy was developed to run realistic simulations only with public transport but no
cars. With the results of a relaxed simulation of the MATSim Singapore scenario including both
public and private transport, the mean and standard deviation of the travel times between two
consecutive stops were saved for individual time bins of constant duration over the course of a
day. Trading off memory consumption against the ability to still allow for dynamic travel time
variation, we use time bins of 15 minutes. Then a simplified road network was created with just
one link between two consecutive public transport stops as shown in Figure 3. The speed of the
vehicles in these links is derived in a stochastic way assuming a normal distribution with the
parameters previously saved. To avoid negative travel times the normal distribution was cut off
for travel speeds below 10 km/h. Generally, this type of distribution being used can easily be
changed, e.g. to a Burr type XII distribution as suggested by Susilawati and Taylor (16). With
this strategy the effect that other vehicles have on buses is simplified and mobility simulation
times are reduced by 60%. Dwelling processes will still depend on the demand.

**RESULTS**

**Functional results**

**Relaxation process**

As it was mentioned before, MATSim uses an evolutionary algorithm to improve agents’
experience until the simulation reaches user equilibrium. The number of iterations needed to
reach this state is a critical variable and many efforts have been performed to reduce it. (17),
(18). The EBR effectively reduces the number of iterations public transport users need to reach
equilibrium. Figure 4 shows average scores of the executed plans of the 355,207 agents trough
100 iterations. Agents are saving five plans in memory, and the graph shows the average score
of the executed plans, the average score of the best and worst performing plan as well as the
average of the averaged score of the five plans. At iteration 0, both EBR and SBR start with
routes as described in the schedule. However, the EBR returns routes which perform better
in the simulation. The cause of this difference is related with the time the routers take into
account as travel time between two consecutive stops. The EBR uses the average between all the
scheduled times of the routes which contain these two consecutive stops. This means that the
EBR does not take into account specific travel times for each service between two consecutive
stops but an average within a given time bin. The SBR uses the specific scheduled time of the
corresponding route. According to the results the average seems to be a more reliable estimate
for the first iteration, maybe because the specific route offsets are a very rough quantity (in the
To exemplify the functional and performance difference of the two routers, we present in Figure 4 a comparison of score evolution for two replanning strategies. The first strategy stipulates that 30% of the agents are re-routed at each iteration.

Using SBR, agents receive the same route over and over again with the SBR as the start time origin and destination do not change between iterations. Small variations in scores occur because of the stochastic nature of the simulation explained above. Starting in the initial iteration with a score in the same range as SBR, using EBR caused that better performing routes are found within a very small number of iterations. Furthermore, as the difference in the score between the best and the worst averages demonstrates, agents save plans with alternative routes.

For a more realistic comparison the 355,207 agents are simulated with both routers for 100 iterations again but just 20% of them are re-routed and the start times are modified randomly within a half an hour for 10% of them. The second graph shows how both routers manage to improve agents’ plans. But with the EBR the number of iterations needed to achieve the average executed score achieved after 100 iterations for the SBR (120) was just 5. Another impressive conclusion is that with the replanning strategy of 30% re-routing only, this number is achieved after only 3 iterations. The marginal score (as a measurement of relaxation advance) after 200 iterations with the SBR (0.1 utility point) is achieved after 77 iterations with the EBR. This means an improvement by a factor of 2.6.

**Modeling advantages**

As the disutility function of the links in the proposed network account for aspects like waiting times or occupancy levels, and as MATSim allows to model heterogeneity among agents, the case of Singapore these scheduled values are given in minutes).
FIGURE 4  Comparison of score evolution: a) 30% Re-route, b) 20% Re-route and 10% Time allocation
router results in a very powerful tool to model emergent behaviour in public transport route choice as observed in the reality. In Singapore, as many other crowded cities in the world, some people decide to travel backwards for a few stops and transfer to a train in the opposite direction to find a seat or space in a public transport vehicle (19). With the SBR it is not possible that a least cost path of such type is found, but with the new proposal this is possible. Although proportions don’t match with actual observations as we are lacking appropriate and calibrated utility parameters for traveling and waiting time under crowded conditions, Figure 5 shows totals of people traveling backwards from different stops in the island after 100 iterations (see Figure 4 (a)).

As shown in the previous section, another important quality is that the EBR is capable to create alternative routes even with the same start time, origin and destination.

Comparing quality attributes with the current implementation

Computation time

The tests described next were executed on a using 12 computational nodes accessing 70GB of shared memory using the Singapore scenario described in the previous section. Before the first iteration, if plans are not routed, MATSim prepares every agent with an initial route. As it was mentioned before, the stop-stop travel times and stop-route waiting times are initially taken from the schedule. Because of its simpler network structure the EBR takes 01:17:35 to initially route the 355,207 users compared with 01:28:55 needed by the SBR which equals to a gain of about 12.7% for this scenario. When running MATSim iterations with the EBR, computation times principally change in two processes: mobility simulation (mobsim) and replanning. Figure 6 shows computation times measured for the first 20 iterations of the process. Although the EBR needs more time in mobsim, it continues to require considerably less time for
re-routing during the replanning due to a simpler network topology. The longer mobsim time is due to the saving of information in the mentioned structures during the course of the simulation. However, in average the EBR outperforms SBR per iteration by about 3 minutes or 11%. As mentioned above 2.6 times more iterations are needed for the SBR to achieve a specific point in the relaxation process. For 77 iterations with the EBR, the computation amounts 35:25:43, and for 200 iterations with the SBR the computation amounts 99:10:51; that is an improvement by a factor of 2.8 in our experimental setting.

Memory consumption

The EBR needs more memory than the SBR because the ERB manages more information. The necessary extra memory is allocated to the three structures described before. Given the conditions described for the Singapore scenario the extra memory is calculated as follows. One numeric value needs 8 Bytes, and with a time bin of 15 minutes, 120 bins are needed for 30 hours. The Stop-stop travel times structure saves two values (average and number of observations) per each time bin per each pair of consecutive stops. The number of pairs for the Singaporean public transport system is 6,602. Thus, this structure approximately needs 12.7 MB. Similarly, The stop-route waiting time structure saves two values (average and number of observations) per each time bin per each pair of stop route combinations. The number of stop-route relations for the Singaporean public transport system is 27,156. Thus, this structure approximately needs 52.1 MB. Finally the vehicle occupancy structure saves the average and the number of observations for 26,353 Route-stop relations for each of the 120 time bins. Hence it requires approximately 50.7 MB. In total less than 120 MB are needed for the three structures.
On the other hand the size of the network where public transport routes are calculated is smaller for the EBR. Although, in the case of Singapore, it creates 31,939 nodes compared with 27,156 of the SBR (4,783 new stop-nodes) the number of links is dramatically smaller. The SBR creates 424,070 walking links and 26,353 travel links (450,423 in total). The EBR creates the same 26,353 travel links, plus 27,156 boarding links, plus 27,156 alighting links and just 4,390 walking links (85,055 links in total); less than a fifth in total. As a node needs 48 Bytes and a link 128 Bytes approximately the SBR needs 46.8 MB more memory for links and just 229.6 KB less for nodes. The EBR saves 46.5 MB for the network, concluding that in total the SBR needs 70 MB less memory. This quantity in neglectable compared with the total memory needed for the whole process.

CONCLUSION AND FUTURE WORK

In this work a new public transport router for the multi-agent activity-based transport simulator MATSim was designed, implemented and tested. It allows to obtain more diverse routes in large scale scenarios taking into account many complexities of urban public transport systems. On the supply side, the system simulates congestion, occupancy levels of public transport vehicles, queues in public transport stops, bay sizes, and bus or train bunching. On the demand side, in addition to the commonly used factors such as in-vehicle time, number of transfers and walking time, the presented new router takes into account disutility of additional waiting time due to congestion or overcrowded vehicles, comfort level inside public transport vehicles and preference heterogeneity among agents for all mentioned factors.

The usability was tested in a large scale scenario of Singapore. Using a simplified public transport only simulation, 100 iterations of a 25% scenario (355,207 agents) with 30% of the agents re-routing each iteration took just 45 hours approximately, or about 27 minutes per iteration, using 12 cores and 70 GB of memory. If just 20% of the plans are re-routed and using 35 cores accessing 85GB of memory, the time per iteration is reduced to less than 13 minutes, achieving 100 iterations in less than one day. But in terms of computation time gains most importantly, we showed that the proposed events-based router is able to reach a steady in a much smaller number of iterations.

The current scheduled-based router of MATSim would still be relevant to be used if the topology of the network is changed for the proposed one, and for public transport systems which in reality can be operated in a very reliable manner with little situations of overcrowding. In that case routing calculations would be as fast as the events-based router, and the mobility simulation would be faster as no information (in-vehicle time, wait time and occupancy) would be needed.

In terms of the resulting network loading, the potentially biggest advantage of the proposed events-based router is its capacity to generate emergent behaviour in congested public transport systems which is in line to actual observations. Future research should aim at estimating the various route choice behaviour parameters corresponding to the provided functionalities of the proposed system and calibrate the simulation. Although the currently used values came from a stated preference survey commissioned by the Land Transport Authority for the case of Singapore, advanced studies could for example be tailored to quantify preference heterogeneity. Furthermore, results from work in progress about the value of a seat in Singapore and the disutility due to discomfort can improve the prediction confidence. Finally information from the smart card data of Singapore can be used for revealed preference estimation of further behavioural parameters such as the quality of a transfer described e.g. by the number of escalators to further refine the system.
REFERENCES


